

PLEASE **AMEND** THE CLAIMS AS FOLLOWS:

1. (Currently Amended) A computer program product for managing execution of an application according to ~~an application~~ a lifecycle, the computer program product comprising:
 - a computer-readable medium storing computer-readable instructions thereon, the computer-readable instructions including:
 - instructions for receiving a state change request from ~~an~~ the application, the state change request indicating a request from the application that an application manager initiate a change in state of the application from a first state to a second state; and
 - instructions for initiating the state change of the application in response to the state change request when the second state is an allowable state according to a specified set of rules.
2. (Original) The computer program product as recited in claim 1, wherein the second state is an active state indicating that the application is currently executing.
3. (Currently Amended) A computer program product for managing execution of an application according to ~~an application~~ a lifecycle, the computer program product comprising:
 - a computer-readable medium storing computer-readable instructions thereon, the computer-readable instructions including:
 - instructions for receiving a signal indicating that a new service is selected;
 - instructions for initiating execution of the application when the new service is selected such that the application enters an active state;
 - instructions for pausing execution of the application such that the application enters a paused state from the active state;
 - instructions for receiving a resume request from the application indicating that the application wishes to resume execution and enter the active state from the paused state; and
 - instructions for starting execution of the application such that the application enters the active state from the paused state when the resume request is received from the application.
4. (Currently Amended) A computer program product for managing execution of a plurality of applications according to ~~an application~~ a lifecycle, the computer program product comprising:

a computer-readable medium storing computer-readable instructions thereon, the computer-readable instructions including:

instructions for initiating execution of each one of the plurality of applications such that the plurality of applications enter an active state;

instructions for pausing execution of one of the plurality of applications such that the one of the plurality of applications enters a paused state from the active state;

instructions for receiving a resume request from one or more of the plurality of applications indicating that the one or more of the plurality of applications request to resume execution and enter the active state from the paused state;

instructions for selecting one of the one or more of the plurality of applications from which the resume request was received ~~to move from the paused state to the active state~~; and

starting execution of the selected application such that the selected application enters the active state from the paused state in response to receiving the resume request when the resume request is received from the application.

5. (Currently Amended) A computer program product for managing execution of an application according to an application a lifecycle, the computer program product comprising:

a computer-readable medium storing computer-readable instructions thereon, the computer-readable instructions including:

instructions for requesting a first time that an the application change its state from a first state to a second state;

instructions for determining whether the application has changed its state from the first state to the second state; and

instructions for requesting a second time that the application change its state from the first state to the second state when it is determined that the application has not changed its state from the first state to the second state and a predetermined condition is satisfied.

6. (Original) The computer program product as recited in claim 5, wherein the predetermined condition indicates that a specified period of time has elapsed or that the application is now able to perform the requested state change.

7. (Original) The computer program product as recited in claim 5, wherein it is determined that the application has not changed its state when a state change exception is raised by the application.

8. (Original) The computer program product as recited in claim 5, wherein it is determined that the application has not changed its state when the application rejects the requested state change.

9. (Original) The computer program product as recited in claim 5, wherein it is determined that the application has not changed its state when the application is unable to perform the requested state change.

10. (Currently Amended) A computer program product for managing execution of an application according to ~~an application~~ a lifecycle, the computer program product comprising: a computer-readable medium storing computer-readable instructions thereon, the computer-readable instructions including:

instructions for requesting that ~~an~~ the application change its state from a first state to a second state;

instructions for determining whether the application has changed its state from the first state to the second state; and

instructions for requesting that the application change its state from the first state to a third state when it is determined that the application has not changed its state from the first state to the second state.

11. (Original) The computer program product as recited in claim 10, wherein the first state is an active state indicating that the application is currently executing, the second state is a destroyed state indicating that the execution of the application has terminated, and the third state is a paused state indicating that execution of the application has paused such that the application can resume execution.

12. (Original) The computer program product as recited in claim 10, wherein it is determined that the application has not changed its state when a state change exception is raised by the application.

13. (Original) The computer program product as recited in claim 10, wherein it is determined that the application has not changed its state when the application rejects the requested state change.

14. (Original) The computer program product as recited in claim 10, wherein it is determined that the application has not changed its state when the application is unable to perform the requested state change.

15. (Currently Amended) A computer program product for managing execution of an application according to ~~an application~~ a lifecycle, the computer program product comprising:
a computer-readable medium storing computer-readable instructions thereon, the computer-readable instructions including:

instructions for requesting that a first application change its state from a first state to a second state;

instructions for determining whether the first application has changed its state from the first state to the second state; and

instructions for requesting that a second application change its state from the first state to the second state when it is determined that the first application has not changed its state from the first state to the second state.

16. (Original) The computer program product as recited in claim 15, wherein the first state is an active, paused, or loaded state and the second state is a destroyed state indicating that the application is terminated.

17. (Original) The computer program product as recited in claim 15, wherein it is determined that the first application has not changed its state when a state change exception is raised by the first application.

18. (Currently Amended) The computer program product as recited in claim 17, wherein the second state is an active state indicating that the ~~associated~~ application is being executed, and the state change exception is raised by the first application when the first application has entered itself into a paused state or a terminated state.

19. (Original) The computer program product as recited in claim 15, wherein it is determined that the first application has not changed its state when the first application rejects the requested state change.

20. (Original) The computer program product as recited in claim 15, wherein it is determined that the first application has not changed its state when the first application is unable to perform the requested state change.

21. (Currently Amended) A system for managing execution of an application according to ~~an application-a~~ lifecycle, the system comprising:

one or more rules; and

an application manager capable of executing one or more applications according to ~~an the application lifecycle, the lifecycle~~ enabling each of the applications to enter one of a plurality of states in response to one or more associated predetermined commands, the application manager capable of selecting one of the predetermined commands to execute according to the one or more rules.

22. (Original) The system as recited in claim 21, further comprising:

a signaling monitor coupled to the application manager and capable of receiving a data stream, the signal monitor adapted for determining whether an application is present in the data stream and communicating information associated with the application to the application manager.

23. (Original) The system as recited in claim 21, wherein the application manager is configured to store an application context for each of the applications, the application context identifying a current one of the plurality of states.

24. (Original) The system as recited in claim 23, wherein the current one of the plurality of states is identified by the associated application to the application manager.

25. (Original) The system as recited in claim 23, wherein the application context further identifies a class loader capable of loading one or more classes associated with the application.

26. (Original) The system as recited in claim 23, wherein the application context further identifies a display context including display information to be displayed.

27. (Currently Amended) The system as recited in claim 23, wherein the application context further identifies an application environment object enabling the ~~associated~~ application to communicate with the application manager.

28. (Currently Amended) The system as recited in claim 23, wherein the application context further identifies an application environment object that enables the ~~associated~~ application to retrieve properties associated with its runtime environment.

29. (Currently Amended) The system as recited in claim 23, wherein the application context further identifies an application environment object that enables the ~~associated~~ application to communicate a state change of the application to one of the plurality of states.

30. (Currently Amended) The system as recited in claim 23, wherein the application context further identifies an application environment object that enables the ~~associated~~ application to request that the application manager change the current state of the application from a paused state to an active state.

31. (Original) The system as recited in claim 21, further comprising:
a display manager coupled to the application manager and adapted for managing a display context for each of the applications, the display context being in a first state when the display context is visible and being in a second state when the display context is invisible.

32. (Currently Amended) The system as recited in claim 31, wherein the display context is in the first state when the ~~associated~~-application is in an active state and in the second state when the ~~associated~~ application is in a paused state.

33. (Original) The system as recited in claim 31, wherein the state of the display context is determined according to the one or more rules followed by the application manager.

34. (Currently Amended) A digital television receiver for managing execution of an application according to ~~an application~~ a lifecycle, comprising:
a processor; and
instructions including:
instructions for determining from a data stream whether an application is present in the data stream;
instructions for loading ~~the~~ an application when it is determined that ~~an~~ ~~the~~ application is present in the data stream; and

instructions for executing the loaded application according to ~~an application~~ the lifecycle, the lifecycle including a plurality of states.

35. (Currently Amended) The digital television receiver as recited in claim 34, wherein the instructions for executing the application comprise:

a first interface that is visible to an application manager, the first interface adapted for enabling the application manager to cause the application to change from one of the plurality of states to another one of the plurality of states; and

a second interface that is visible to the application, the second interface adapted for enabling the application to communicate to the application manager a state change of the application from one of a first set of the plurality of states to one of a second set of the plurality of states.

36. (Original) The digital television receiver as recited in claim 35, wherein the second set of the plurality of states includes a paused state indicating that the application has been paused and a destroyed state indicating that the application has been terminated.

37. (Original) The digital television receiver as recited in claim 34, wherein the instructions for executing the application comprise:

a first interface that is visible to an application manager, the first interface adapted for enabling the application manager to cause the application to change its state from one of the plurality of states to another one of the plurality of states; and

a second interface that is visible to the application, the second interface adapted for enabling the application to request that the application manager change the state of the application to a first one of the plurality of states.

38. (Original) The digital television receiver as recited in claim 37, further comprising: instructions for changing the state of the application from a second one of the plurality of states to the first one of the plurality of states.

39. (Original) The digital television receiver as recited in claim 38, wherein the first state is an active state and the second state is a paused state.

40. (Original) The digital television receiver as recited in claim 34, wherein the instructions for executing the application comprise:

a first interface that is visible to an application manager, the first interface adapted for enabling the application manager to cause the application to change its state from one of the plurality of states to another one of the plurality of states; and

a second interface that is visible to the application, the second interface adapted for enabling the application to communicate to the application manager that the application cannot change its state as the application manager has requested.

41. (Original) The digital television receiver as recited in claim 40, further comprising: instructions enabling the application to raise a state change exception indicating that the application cannot change its state as the application manager has requested.

42. (Original) The digital television receiver as recited in claim 40, further comprising: instructions enabling the application to raise a state change exception indicating that the application does not want to change its state as the application manager has requested.

43. (Original) The digital television receiver as recited in claim 36, further comprising: instructions for releasing memory associated with the application when the application has been terminated.

44. (Original) The digital television receiver as recited in claim 34, further comprising: instructions for creating a class loader associated with the application, the class loader being adapted for loading one or more classes associated with the application; instructions for employing the class loader to load the classes associated with the application; and instructions for instantiating the application using the classes that have been loaded by the class loader.

45. (Original) The digital television receiver as recited in claim 44, further comprising: instructions for unloading the classes associated with the application when the application is terminated.

46. (Original) The digital television receiver as recited in claim 45, wherein the instructions for unloading the classes comprise: instructions for de-referencing the class loader.